**資料庫設定**

**變更上傳到 GitHub**

**流程就是：add、commit 和 push。**

**重複上傳的完整流程**

**假設你已經在本地的 index.html 中新增了一些功能或修改了文字，以下是具體的指令：**

**步驟一：加入變更到暫存區 (git add)**

**首先，告訴 Git 你要追蹤這次的變更。**

**Bash**

**git add index.html**

**或者，如果你想將所有修改過的檔案都加入，可以用這個指令：**

**Bash**

**git add .**

**步驟二：提交變更到本地倉庫 (git commit)**

**接著，將這些變更提交到你的本地歷史記錄中，並附上一段有意義的訊息，說明你做了什麼修改。這是一個好習慣，可以讓你和團隊成員日後更容易回溯版本。**

**Bash**

**git commit -m "Updated the UI and fixed a bug in the TTS feature"**

**git commit -m "feat: Add playback speed control to podcast app"**

**這段訊息應該要具體，例如 "修復了播放按鈕的錯誤" 或 "新增了新的 UI 元素"。**

**步驟三：推送到遠端倉庫 (git push)**

**最後，將你的本地提交推送到 GitHub 的遠端倉庫。**

**Bash**

**git push**

**由於你之前已經設定了 -u origin main，Git 現在知道要將變更推送到哪裡，所以你不需要再指定遠端和分支名稱了。**

**自動化部署的好處**

**當你完成這三個步驟並成功 push 之後，Netlify 會自動偵測到 GitHub 上的變更，並立即開始重新部署你的網站。這就是自動化部署的強大之處：**

* **自動偵測：你不需要手動上傳檔案。**
* **快速部署：變更通常在幾秒鐘到幾分鐘內就能上線。**
* **版本控制：你的所有變更都有記錄，方便你隨時回溯到任何一個舊版本。**

**因此，只要你記得這個 add -> commit -> push 的循環，就可以輕鬆地管理和更新你的網站了。**

讀取舊的-刪掉最新的—回到前一版

**最好的做法：使用 git revert**

正如我們之前討論的，如果你已經將更改推送到遠端倉庫（GitHub），我強烈建議你使用 git revert。這個指令會創建一個**新的提交**來撤銷前一個提交的變更，這樣就不會影響歷史記錄，對與團隊協作的專案非常重要。

Bash

git revert HEAD

這個指令會自動生成一個新的提交，將你最新的更改撤銷。然後，你只需要將這個新的提交推送到遠端即可。